DETECTING FAKE AND PHISHING WEBSITES

SYNOPSIS/PROPOSAL

Submitted By:

Anukriti (2K17/IT/027)

Anurag Mudgil (2K17/IT/029)

PROBLEM STATEMENT

Phishing is one of the luring techniques used by phishing artists with an intention of exploiting the personal details of unsuspected users. Generally in a phishing attack, emails, messages etc are sent to the user claiming to be a legitimate organization, asking the user to enter their personal information by redirecting them to some fake website (phishing website) that looks similar in appearance but different in destination. Through these websites, the phishers tend to steal the user's information and thus, carry out illegal transactions. Detection of these websites are still a big concern for cyber security to save millions of users from falling into the trap of phishing and hence, there is a need for an efficient mechanism for the prediction of phishing websites.

PLAN OF EXECUTION (PROPOSAL):

In this project, we intend to address the problem of detection of malicious URLs as a multi-class classification problem in which we will classify the raw URLs into different categories such as benign or safe URL, phishing URL, malware URL or defacement URL.

❖ COLLECTION OF DATA (Dataset to be used):

Data will be aggregated from 5 different sources as mentioned below:

- **Phishing Websites Data Set** (from UCI Machine Learning Repository)
- **URL 2016 | Datasets | Research | Canadian Institute for Cybersecurity** (URL dataset (ISCX-URL2016)
- **Developers** (Phishtank dataset)
- **PhishStorm - phishing / legitimate URL dataset — Aalto University** (PhishStorm dataset)
- **DNS-BH – Malware Domain Blocklist by RiskAnalytics » BH DNS Files** (Malware domain black list dataset)

Description of the final dataset obtained:

For training and testing machine learning algorithms, we have planned to form a dataset with URLs having different categories: benign/safe, defacement, phishing and malware. For collecting these categories, we will be using the URL dataset (ISCX-URL-2016). For increasing phishing and malware URLs, we will be using Malware domain black list dataset. At last, we have increased the number of phishing URLs using the Phishtank dataset and PhishStorm dataset. Firstly, we will be collecting the URLs from different sources into a separate data frame and then will finally merge them to retain only URLs and their class type.

❖ FEATURE ENGINEERING:

As we know machine learning algorithms only support numeric inputs so we will create lexical numeric features from input URLs. So the input to machine learning algorithms will be the numeric lexical features rather than actual raw URLs.

Feature Selection:

The dataset from the UCI - Machine Learning Repository consists of the URL information using 30 features but out of these features, it was infeasible to extract all the features because many features used some standard databases which are not accessible to us. Also, extracting some of the features seemed not possible as they demand the extraction of data from the server of the website, which is not possible. Hence, we narrowed down our dataset to contain 22 features.

These are: having_ip_address, url_length, shortening_service, having_at_symbol, https_token, favicon, double_slash_redirecting, prefix_suffix, having_sub_domain, domain_registration_length, sfh, url_of_anchor, request_url, url, links_in_tags, submitting_to_email, abnormal_url, sus_domain, age_of_domain, i_frame, dns, web_traffic, google_index, statistical_report.

Grid Search/Random Search:

We will also get features importance using gridsearchcv/randomizedsearchcv to obtain best parameters and fit the model with best parameters to obtain a better accuracy, if needed.

❖ CHOICE OF MODEL/CLASSIFICATION ALGORITHM:

- ❖ Some of the classification algorithms that we were considering initially were: Naive Bayes, J48 (Decision Tree), IBK and SVM but on the basis of our study from **Comparison of TP Rate, FP Rate and Detection Accuracy**, we discarded SVM, IBK and Naive Bayes because of their low accuracy, precision, recall and F1-Score.
- ❖ We will be considering some well-known boosting machine learning classifiers as well namely XGBoost, Light GBM, Gradient Boosting Machines.
- ❖ On studying **Malicious web content detection using machine leaning - IEEE Conference Publication**, we studied about the random forest classifier and found out that it will be a better classifier than decision tree model as it considers the combination of various tree predictors.

- We will be using Ensemble techniques using Random forest classifiers along with boosting machine learning algorithms.

❖ PERFORMANCE METRICS:

Combination of below mentioned metrics:

- Accuracy
- Precision
- F1-Score
- Recall
- Area under ROC curve

TECHNOLOGY STACK:

- **MACHINE LEARNING CLASSIFIER:**
  - **Libraries:** pandas, itertools, bs4 (BeautifulSoup), sklearn (classifiers, metrics, model_selection), matplotlib (visualization) etc.
  - **Tools:** Anaconda, WEKA
  - **Language:** Python

- **GUI/DEMO:**
  - **Frontend: React**
  - **Backend: Flask**
  - **Languages: Javascript, Python**

GUI/DEMO:

We will be creating a simple web application where a user can enter the URL as an input (for detecting whether it is a phishing or benign URL, this data will then pass through our model and finally, will output the result “SAFE/UNSAFE” to the user. This will prevent them from any phishing attacks as before clicking on any URL, they will always have an option to find out if its a phishing website and hence, can safeguard themselves from any future attack.